Refinement Strategies for Verification Methods Based on Datapath Abstraction

Zaher S. Andraus, Mark H. Liffiton, & Karem A. Sakallah
University of Michigan

ASP-DAC 2006
Yokohama, Japan
January 25, 2006
REVEAL

- Formal equivalence verification of Verilog designs
- Based on automatic *datapath abstraction & refinement*
- Leverages recent advances in *scalable automated reasoning methods*
- Suitable for verification of (control logic of) pipelined microprocessors against their ISA specs
- Extensible (through suitable definition of “equivalence”) to other high-performance microarchitectures
Basic CEGAR

conc(X)

relax

abst(X) = abst(X) && !X*

abst(X) → prop(X)

Counter
Example X*

conc(X*)

Prop Holds

Prop Fails
CEGAR with Enhanced Refinement

conc(X)

relax

expl(X) = MUS(conc(X) && viol(X))
abst(X) = abst(X) && !(viol(X) && expl(X))

abst(X)

abst(X) → prop(X)

Counter Example X*

viol(X) = localize(X*)
viol(X) = generalize(viol(X))
expl(X) = MUS(!abst(X) && viol(X))
viol(X) = viol(X) && expl(X)

Prop Holds

conc(X) && viol(X)

Prop Fails
Property Satisfied by Abstract Model

- prop
- abst
- conc
Property Violated by Abstract Model but Satisfied by Concrete Model
Property Violated by both Abstract and Concrete Models
Datapath Abstraction
Datapath Abstraction
module example();

wire [3:0] a, b;
wire m =a[3]; // msb
wire l = a[0]; // lsb
wire c = m? a >> 1 : a;
wire d = l? b >> 2 : c;
wire e = m? a : a >> 1;
wire f = l? {2’b00, b[3:2]} : e;
wire p = !(a == 0) || (d == f);
endmodule

cncc ( a,b,c,d,e,f,l,m,p ) =
(m = a[3]) ∧
(l = a[0]) ∧
(m ∧ ( c = a >> 1) ∨ ¬m ∧ ( c = a)) ∧
(l ∧ ( d = b >> 2) ∨ ¬l ∧ ( d = c)) ∧
(m ∧ ( e = a) ∨ ¬m ∧ ( e = a >> 1)) ∧
(l ∧ ( f = { 2'b00, b[3 : 2]}) ∨ ¬l ∧ ( f = e )) ∧
(p = ¬( a = 0) ∨ ( d = f ))
Running Example

\[ \text{conc}(a,b,c,d,e,f,l,m,p) = \]
\[ (m = a[3]) \land \]
\[ (l = a[0]) \land \]
\[ (m \land (c = a \gg 1) \lor \neg m \land (c = a)) \land \]
\[ (l \land (d = b \gg 2) \lor \neg l \land (d = c)) \land \]
\[ (m \land (e = a) \lor \neg m \land (e = a \gg 1)) \land \]
\[ (l \land (f = \{2'b00, b[3:2]\}) \lor \neg l \land (f = e)) \land \]
\[ (p = \neg(a = 0) \lor (d = f)) \]

\[ \text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,zero) = \]
\[ (m = \text{EX1}(a)) \land \]
\[ (l = \text{EX2}(a)) \land \]
\[ (s = \text{SR1}(a,\text{succ}(zero))) \land \]
\[ (t = \text{SR2}(b,\text{succ}(\text{succ}(zero)))) \land \]
\[ (u = \text{CT1}(zero, \text{EX3}(b))) \land \]
\[ (c = \text{ite}(m,s,a)) \land \]
\[ (d = \text{ite}(l,t,c)) \land \]
\[ (e = \text{ite}(m,a,s)) \land \]
\[ (f = \text{ite}(l,u,e)) \land \]
\[ (p = \neg(a = 0) \lor (d = f)) \land \]

Functional Consistency Constraints
Functional Consistency

\[(y_1 = x_1) \land (y_2 = x_2) \rightarrow F(y_1, y_2) = F(x_1, x_2)\]
**1st Iteration: Counterexample**

<table>
<thead>
<tr>
<th>Condition Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abst(a,b,c,d,e,f,l,m,s,t,u,p,zero) =</td>
<td>Functional Consistency Constraints</td>
</tr>
<tr>
<td>(m = EX1(a)) ∧</td>
<td></td>
</tr>
<tr>
<td>(l = EX2(a)) ∧</td>
<td></td>
</tr>
<tr>
<td>(s = SR1(a,succ(zero))) ∧</td>
<td></td>
</tr>
<tr>
<td>(t = SR2(b,succ(succ(zero)))) ∧</td>
<td></td>
</tr>
<tr>
<td>(u = CT1(zero,EX3(b))) ∧</td>
<td></td>
</tr>
<tr>
<td>(c = ITE(m,s,a)) ∧</td>
<td></td>
</tr>
<tr>
<td>(d = ITE(l,t,c)) ∧</td>
<td></td>
</tr>
<tr>
<td>(e = ITE(m,a,s)) ∧</td>
<td></td>
</tr>
<tr>
<td>(f = ITE(l,u,e)) ∧</td>
<td></td>
</tr>
<tr>
<td>(p = ¬(a = 0) ∨ (d = f)) ∧</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Condition Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abst(a,b,c,d,e,f,l,m,s,t,u,p,zero) =</td>
<td></td>
</tr>
<tr>
<td>(1 = EX1(0)) ∧</td>
<td></td>
</tr>
<tr>
<td>(1 = EX2(0)) ∧</td>
<td></td>
</tr>
<tr>
<td>(16 = SR1(0,succ(zero))) ∧</td>
<td></td>
</tr>
<tr>
<td>(20 = SR2(8,succ(succ(zero)))) ∧</td>
<td></td>
</tr>
<tr>
<td>(12 = CT1(zero,EX3(8))) ∧</td>
<td></td>
</tr>
<tr>
<td>(16 = ITE(116,0)) ∧</td>
<td></td>
</tr>
<tr>
<td>(20 = ITE(120,16)) ∧</td>
<td></td>
</tr>
<tr>
<td>(0 = ITE(10,16)) ∧</td>
<td></td>
</tr>
<tr>
<td>(12 = ITE(112,0)) ∧</td>
<td></td>
</tr>
<tr>
<td>(0 = ¬(0 = 0) ∨ (20 = 12)) ∧</td>
<td></td>
</tr>
</tbody>
</table>

**Ex. 1:**

**Ex. 2:**

---

**SR1 , zero**

**SR2 , zero**

**CT1 zero,EX3**
1\textsuperscript{st} Iteration: Localize Counterexample

\[
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) = \\
(m = \text{EX1}(a)) \land \\
(l = \text{EX2}(a)) \land \\
(s = \text{SR1}(a,\text{succ}(\text{zero}))) \land \\
(t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land \\
(u = \text{CT1}(\text{zero},\text{EX3}(b))) \land \\
(c = \text{ite}(m,s,a)) \land \\
(d = \text{ite}(l,t,c)) \land \\
(e = \text{ite}(m,a,s)) \land \\
(f = \text{ite}(l,u,e)) \land \\
(p = \neg(a = 0) \lor (d = f)) \land \\
\text{Functional Consistency Constraints}
\]

\[
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) = \\
(1 = \text{EX1}(0)) \land \\
(1 = \text{EX2}(0)) \land \\
(16 = \text{SR1}(0,\text{succ}(\text{zero}))) \land \\
(20 = \text{SR2}(8,\text{succ}(\text{succ}(\text{zero})))) \land \\
(12 = \text{CT1}(\text{zero},\text{EX3}(8))) \land \\
(16 = \text{ite}(1,16,0)) \land \\
(20 = \text{ite}(1,20,16)) \land \\
(0 = \text{ite}(1,0,16)) \land \\
(12 = \text{ite}(1,12,0)) \land \\
(0 = \neg(0 = 0) \lor (20 = 12)) \land \\
\text{Functional Consistency Constraints}
\]
1st Iteration: Generalize Counterexample

\[ \text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) = \]
\[ (m = \text{EX1}(a)) \land \]
\[ (l = \text{EX2}(a)) \land \]
\[ (s = \text{SR1}(a,\text{succ}(\text{zero}))) \land \]
\[ (t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land \]
\[ (u = \text{CT1}(\text{zero},\text{EX3}(b))) \land \]
\[ (c = \text{ite}(m,s,a)) \land \]
\[ (d = \text{ite}(l,t,c)) \land \]
\[ (e = \text{ite}(m,a,s)) \land \]
\[ (f = \text{ite}(l,u,e)) \land \]
\[ (p = \neg(a = 0) \lor (d = f)) \land \]

Functional Consistency Constraints

(a = 0) \land \]
(l = 1) \land \]
(t \neq u)
\[
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) = \\
(m = \text{EX1}(a)) \land \\
(l = \text{EX2}(a)) \land \\
(s = \text{SR1}(a,\text{succ}(\text{zero}))) \land \\
(t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land \\
(u = \text{CT1}(\text{zero},\text{EX3}(b))) \land \\
(c = \text{ite}(m,s,a)) \land \\
(d = \text{ite}(l,t,c)) \land \\
(e = \text{ite}(m,a,s)) \land \\
(f = \text{ite}(l,u,e)) \land \\
p = \neg(a = 0) \lor (d = f) \land \\
\text{Functional Consistency Constraints}
\]
1st Iteration: Minimal Concrete Explanations

\[
\text{conc}(a,b,c,d,e,f,l,m,p) = \\
(\ m = a[3]) \land \\
(\ l = a[0]) \land \\
(\ m \land (c = a \gg 1) \lor \neg m \land (c = a)) \land \\
(l \land (d = b \gg 2) \lor \neg l \land (d = c)) \land \\
(m \land (e = a) \lor \neg m \land (e = a \gg 1)) \land \\
(l \land (f = \{2'b00,b[3:2]\}) \lor \neg l \land (f = e)) \land \\
(p = \neg(a = 0) \lor (d = f)) \\
\]

\[
\text{viol}_1 = (a = 0) \land (a[0] = 1) \\
\text{viol}_2 = b \gg 2 \neq \{2'b00,b[3:2]\}
\]
2\textsuperscript{nd} Iteration: Counterexample

\begin{align*}
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) &= \\
&= (m = \text{EX1}(a)) \land \\
&\quad (l = \text{EX2}(a)) \land \\
&\quad (s = \text{SR1}(a,\text{succ}(\text{zero}))) \land \\
&\quad (t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land \\
&\quad (u = \text{CT1}(\text{zero},\text{EX3}(b))) \land \\
&\quad (c = \text{ite}(m,s,a)) \land \\
&\quad (d = \text{ite}(l,t,c)) \land \\
&\quad (e = \text{ite}(m,a,s)) \land \\
&\quad (f = \text{ite}(l,u,e)) \land \\
&\quad (p = \neg(a = 0) \lor (d = f)) \land \\
&\quad \neg((a = 0) \land (l = 1)) \land \neg(t \neq u)
\end{align*}

Functional Consistency Constraints
2\textsuperscript{nd} Iteration: Localize Counterexample

\begin{align*}
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) &= \\
(m = \text{EX1}(a)) \land \\
(l = \text{EX2}(a)) \land \\
(s = \text{SR1}(a,\text{succ}(\text{zero}))) \land \\
(t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land \\
(u = \text{CT1}(\text{zero},\text{EX3}(b))) \land \\
(c = \text{ite}(m,s,a)) \land \\
(d = \text{ite}(l,t,c)) \land \\
(e = \text{ite}(m,a,s)) \land \\
(f = \text{ite}(l,u,e)) \land \\
(p = \neg(a = 0) \lor (d = f)) \land \\
\neg((a = 0) \land (l = 1)) \land \neg(t \neq u)
\end{align*}

Functional Consistency Constraints

\begin{align*}
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) &= \\
(1 = \text{EX1}(0)) \land \\
(0 = \text{EX2}(0)) \land \\
(8 = \text{SR1}(0,\text{succ}(\text{zero}))) \land \\
(3 = \text{SR2}(16,\text{succ}(\text{succ}(\text{zero})))) \land \\
(3 = \text{CT1}(\text{zero},\text{EX3}(16))) \land \\
(8 = \text{ite}(18,0)) \land \\
(8 = \text{ite}(0,3,8)) \land \\
(0 = \text{ite}(10,8)) \land \\
(0 = \text{ite}(0,3,0)) \land \\
(0 = \neg(0 = 0) \lor (8 = 0)) \land \\
\neg((0 = 0) \land (0 = 1)) \land \neg(3 \neq 3)
\end{align*}

Functional Consistency Constraints
2\textsuperscript{nd} Iteration: Generalize Counterexample

\texttt{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\texttt{zero}) = \\
(m = \texttt{EX1}(a)) \land \\
(l = \texttt{EX2}(a)) \land \\
(s = \texttt{SR1}(a,\texttt{succ}\texttt{(zero)})) \land \\
(t = \texttt{SR2}(b,\texttt{succ}\texttt{(succ}\texttt{(zero)}))) \land \\
(u = \texttt{CT1}(\texttt{zero},\texttt{EX3}(b))) \land \\
(c = \texttt{ite}(m,s,a)) \land \\
(d = \texttt{ite}(l,t,c)) \land \\
(e = \texttt{ite}(m,a,s)) \land \\
(f = \texttt{ite}(l,u,e)) \land \\
(p = \neg((a = 0) \lor (d = f))) \land \\
\neg((a = 0) \land (l = 1)) \land \neg(t \neq u)

Functional Consistency Constraints

\begin{align*}
(a = 0) \land \\
(l = 0) \land \\
(m = 1) \land \\
(t = u) \land \\
(a \neq s)
\end{align*}
2\textsuperscript{nd} Iteration: Minimal Abstract Explanation

\[
\text{abst}(a,b,c,d,e,f,l,m,s,t,u,p,\text{zero}) =
\]
\[
(m = \text{EX1}(a)) \land
\]
\[
(l = \text{EX2}(a)) \land
\]
\[
(s = \text{SR1}(a,\text{succ}(\text{zero}))) \land
\]
\[
(t = \text{SR2}(b,\text{succ}(\text{succ}(\text{zero})))) \land
\]
\[
(u = \text{CT1}(\text{zero},\text{EX3}(b))) \land
\]
\[
(c = \text{ite}(m,s,a)) \land
\]
\[
(d = \text{ite}(l,t,c)) \land
\]
\[
(e = \text{ite}(m,a,s)) \land
\]
\[
(f = \text{ite}(l,u,e)) \land
\]
\[
p = \neg(a = 0) \lor (d = f) \land
\]
\[
\neg((a = 0) \land (l = 1)) \land \neg(t \neq u)
\]

Functional Consistency Constraints

\[
(a = 0) \land
\]
\[
(l = 0) \land
\]
\[
(t = u) \land
\]
\[
(a \neq s)
\]
$\text{conc}(a,b,c,d,e,f,l,m,p) =$

$(m = a[3]) \land$

$(l = a[0]) \land$

$(m \land (c = a \gg 1) \lor \neg m \land (c = a)) \land$

$(l \land (d = b \gg 2) \lor \neg l \land (d = c)) \land$

$(m \land (e = a) \lor \neg m \land (e = a \gg 1)) \land$

$(l \land (f = \{2'b00,b[3:2]\}) \lor \neg l \land (f = e)) \land$

$(p = \neg(a = 0) \lor (d = f))$

$\text{viol}_3 = (a = 0) \land (a \neq a \gg 1)$
## Preliminary Evaluation: Benchmarks

<table>
<thead>
<tr>
<th>Module</th>
<th>Source</th>
<th>Lines</th>
<th>Latches</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLX</td>
<td>VIS</td>
<td>686</td>
<td>396</td>
</tr>
<tr>
<td>Risc16f84</td>
<td>OpenCores</td>
<td>1719</td>
<td>8312</td>
</tr>
<tr>
<td>SimplePipeline</td>
<td>UM</td>
<td>148</td>
<td>1597</td>
</tr>
<tr>
<td>SimpleCorr</td>
<td>UM</td>
<td>86</td>
<td>5</td>
</tr>
</tbody>
</table>
## Preliminary Evaluation: Verification Results

<table>
<thead>
<tr>
<th>Module</th>
<th>Test</th>
<th>Result</th>
<th>CE Length</th>
<th>Run Time (s)</th>
<th>Refinement Iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLX</td>
<td>1</td>
<td>BUG</td>
<td>11</td>
<td>19.13</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>VALID</td>
<td>-</td>
<td>20.91</td>
<td>2</td>
</tr>
<tr>
<td>Risc16f84</td>
<td>1</td>
<td>BUG</td>
<td>4</td>
<td>0.31</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>BUG</td>
<td>4</td>
<td>36.52</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>VALID</td>
<td>-</td>
<td>9.3</td>
<td>1</td>
</tr>
<tr>
<td>SimplePipeline</td>
<td>1</td>
<td>VALID</td>
<td>-</td>
<td>3.2</td>
<td>2</td>
</tr>
<tr>
<td>SimpleCorr</td>
<td>1</td>
<td>VALID</td>
<td>-</td>
<td>2.7</td>
<td>4</td>
</tr>
</tbody>
</table>
Other Refinement Strategies

- Bit Slicing: admit “mistake” and blast terms back to bits
- Find multiple explanations of the violation from the abstract model
- Generalize (and store) inferred “facts”:

\[
\text{Viol} = \neg [CT2(\text{zero,EX4(pc)}) = \text{EX4(pc)}]
\]

\[
\text{Viol}' = \neg [\forall T \cdot CT2(\text{zero},T) = T]
\]
Extensions: Apply on Real-World Examples

- Broader category of pipelines and architectures
  - Out-of-Order Execution
  - Superscalar / VLIW
  - External and Internal Exceptions (Interrupts)
  - Speculation
  - Hierarchical Memory
- Other abstraction schemes to prevent blow-up of CLU formulas
  - Predicate abstraction
  - Temporal abstraction